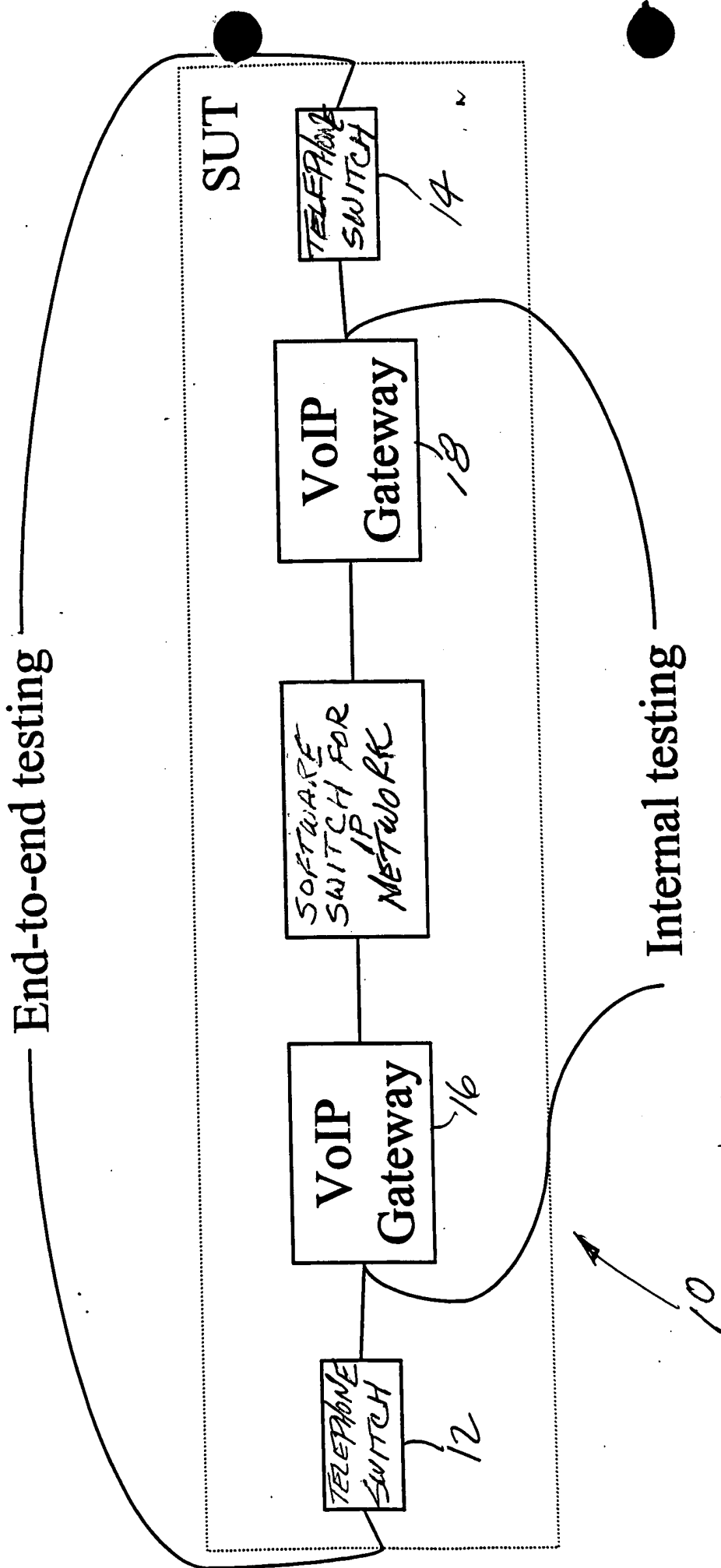
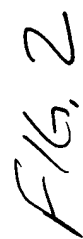


End-to-end testing



Internal testing

FIG. 1



03

gui efsm

states { 21

1. CallingACalledB
2. DialingADialingB
3. IdleABusyB
4. CalledACallingB
5. BusyABusyB
6. DialingALimboB
7. BusyADialingB
8. BusyAIdleB
9. BusyALimboB
10. LimboADialingB
11. ConnectedCalledAConnectedCallerB
12. LimboABusyB
13. IdleADialingB
14. LimboAIdleB
15. DialingABusyB
16. HoldCalledAHoldCallerB
17. HoldCallerAHoldCalledB
18. DialingAIdleB
19. IdleALimboB
20. ConnectedCallerAConnectedCalledB
21. IdleAIdleB

start {
IdleAIdleB

boolean term=false;
boolean IdleA=true;
boolean IdleB=true;
boolean DialingA=false;
boolean DialingB=false;

}

transitions { 68

1. BusyADialingB BusyAIdleB {

i { {On-hook B} }

o { {} }

p { true }

a { { DialingB=false; IdleB=true } }

}

2,

IdleADialingB DialingADialingB {

i { {Off-hook A} }

o { {DialTone A} }

p { true }

a { {IdleA=false; DialingA=true} }

}

3,

BusyADialingB IdleADialingB {

i { {On-hook A} }

o { {} }

p { true }

a { { IdleA=true } }

}

4,

IdleABusyB IdleAIdleB {

i { {On-hook B} }

o { {} }

p { true }

a { {term=true;IdleB=true} }

i = input

o = output

p = predicate

a = action

Fig. 3

```

5. HoldCallerAHoldCalledB IdleAIdleB {
    i { {On-hook A} }
    o { { } }
    p { true }
    a { {term=true; IdleA=true; IdleB=true} }
}

6. IdleALimboB IdleADialingB {
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { { DialingB=true } }
}

7. BusyADialingB BusyABusyB {
    i { {Dial B A} }
    o { {LineBusyTone B} }
    p { true }
    a { { DialingB=false } }
}

8. HoldCalledAHoldCallerB IdleAIdleB {
    i { {On-hook B} }
    o { { } }
    p { true }
    a { {term=true; IdleB=true; IdleA=true} }
}

9. DialingAIdleB DialingADialingB {
    i { {Off-hook B} }
    o { {DialTone B} }
    p { true }
    a { { IdleB=false; DialingB=true } }
}

10. IdleADialingB IdleAIdleB {
    i { {On-hook B} }
    o { { } }
    p { true }
    a { {term=true; IdleB=true; DialingB=false} }
}

11. BusyABusyB BusyAIdleB {
    i { {On-hook B} }
    o { { } }
    p { true }
    a { { IdleB=true } }
}

12. DialingADialingB BusyADialingB {
    i { {Dial A B} }
    o { {LineBusyTone A} }
    p { true }
    a { { DialingA=false } }
}

13. IdleAIdleB IdleADialingB {
    i { {Off-hook B} }
    o { {DialTone B} }
    p { !term }
    a { { IdleB=false; DialingB=true } }
}

14. HoldCalledAHoldCallerB IdleADialingB {
    i { {Timeout AB} }
    o { {DialTone B} }
    p { true }
    a { { IdleA=true; DialingB=true } }
}

15. IdleABusyB DialingABusyB {
    i { {Off-hook A} }
    o { {DialTone A} }
    p { true }
    a { { IdleA=false; DialingA=true } }
}

16. CallingACalledB ConnectedCallerAConnectedCalledB {
    i { {Off-hook B} }
    o { { } }
    p { true }
    a { { } }
}

```

PK. 4

```

)
DialingADialingB DialingABusyB {
  i { {Dial B A} }
  o { {LineBusyTone B} }
  p { true }
  a { { DialingB=false } }
}

17. CalledACallingB IdleALimboB {
  i { {Timeout AB} }
  o { {} }
  p { true }
  a { { IdleA=true } }
}

18. ConnectedCallerAConnectedCalledB IdleALimboB {
  i { {On-hook A} }
  o { {} }
  p { true }
  a { { IdleA=true } }
}

19. ConnectedCallerAConnectedCalledB HoldCallerAHoldCalledB {
  i { {On-hook B} }
  o { {} }
  p { true }
  a { {} }
}

20. IdleAIdleB DialingAIdleB {
  i { {Off-hook A} }
  o { {DialTone A} }
  p { !term }
  a { {IdleA=false; DialingA=true} }
}

21. IdleALimboB IdleAIdleB {
  i { {On-hook B} }
  o { {} }
  p { true }
  a { {term=true; IdleB=true} }
}

22. DialingABusyB BusyABusyB {
  i { {Dial A B} }
  o { {LineBusyTone A} }
  p { true }
  a { { DialingA=false } }
}

23. HoldCalledAHoldCallerB ConnectedCalledAConnectedCallerB {
  i { {Off-hook A} }
  o { {} }
  p { true }
  a { {} }
}

```

```

26. DialingABusyB IdleABusyB (
    i { {On-hook A} }
    o { { } }
    p { true }
    a { { DialingA=false; IdleA=true } }
)

27. DialingABusyB DialingAIdleB (
    i { {On-hook B} }
    o { { } }
    p { true }
    a { { IdleB=true } }
)

28. HoldCallerAHoldCalledB DialingAIdleB (
    i { {Timeout AB} }
    o { {DialTone A} }
    p { true }
    a { { DialingA=true; IdleB=true } }
)

29. DialingAIdleB CallingACalledB (
    i { {Dial A B} }
    o { {AudibleRinging A,Ringing B} }
    p { true }
    a { {DialingA=false; IdleB=false} }
)

30. IdleADialingB CalledACallingB (
    i { {Dial B A} }
    o { {AudibleRinging B,Ringing A} }
    p { true }
    a { { IdleA=false; DialingB=false } }
)

31. BusyABusyB IdleABusyB (
    i { {On-hook A} }
    o { { } }
    p { true }
    a { { IdleA=true } }
)

32. CalledACallingB IdleAIdleB (
    i { {On-hook B} }
    o { { } }
    p { true }
    a { {term=true;IdleA=true;IdleB=true} }
)

33. ConnectedCalledAConnectedCallerB HoldCalledAHoldCallerB (
    i { {On-hook A} }
    o { { } }
    p { true }
    a { { } }
)

34. BusyAIdleB BusyADialingB (
    i { {Off-hook B} }
    o { {DialTone B} }
    p { true }
    a { { IdleB=false; DialingB=true } }
)

35. LimboAIdleB DialingAIdleB (
    i { {Timeout A} }
    o { {DialTone A} }
    p { true }
    a { { DialingA = true; } }
)

36. IdleALimboB DialingALimboB (
    i { {Off-hook A} }
    o { {DialTone A} }
    p { true }
    a { { IdleA=false;DialingA=true} }
)

37. LimboAIdleB LimboADialingB (
    i { {Off-hook B} }
    o { {DialTone B} }
    p { true }
    a { { IdleB=false;DialingB=true} }
)

```

FIG. 6

```

43. DialingALimboB IdleALimboB (
    i { {On-hook A} }
    o { {} }
    p { true }
    a { {IdleA=true;DialingA=false} }
)

44. BusyALimboB IdleALimboB (
    i { {On-hook A} }
    o { {} }
    p { true }
    a { {IdleB=true} }
)

45. BusyALimboB IdleALimboB (
    i { {On-hook A} }
    o { {} }
    p { true }
    a { {term=true;IdleA=true} }
)

46. LimboADialingB LimboAIdleB (
    i { {On-hook B} }
    o { {} }
    p { true }
    a { {DialingB=false;IdleB=true} }
)

47. LimboABusyB LimboAIdleB (
    i { {On-hook B} }
    o { {} }
    p { true }
    a { {IdleB=true} }
)

48. CalledACallingB ConnectedCalledAConnectedCallerB (
    i { {Off-hook A} }
    o { {} }
    p { true }
    a { {} }
)

49. DialingALimboB BusyALimboB (
    i { {Dial A B} }
    o { {LineBusyTone A} }
    p { true }
    a { {DialingA=false} }
)

```

R/S. 7

```

50. DialingADialingB IdleADialingB (
    i { { On-hook A } }
    o { { } }
    p { true }
    a { { IdleA=true; DialingA=false } }
)

51. LimboADialingB LimboABusyB (
    i { { Dial B A } }
    o { { LineBusyTone B } }
    p { true }
    a { { DialingB=false } }
)

52. CallingACalledB LimboAIdleB (
    i { { Timeout AB } }
    o { { } }
    p { true }
    a { { IdleB=true; } }
)

53. BusyAIdleB DialingAIdleB (
    i { { Timeout A } }
    o { { DialTone A } }
    p { true }
    a { { DialingA=true } }
)

54. DialingADialingB DialingAIdleB (
    i { { On-hook B } }
    o { { } }
    p { true }
    a { { DialingB=false; IdleB=true } }
)

55. BusyADialingB DialingADialingB (
    i { { Timeout A } }
    o { { DialTone A } }
    p { true }
    a { { DialingA=true } }
)

56. DialingADialingB IdleADialingB (
    i { { On-hook A } }
    o { { } }
    p { true }
    a { { DialingB=false; IdleB=true } }
)

57. BusyALimboB DialingALimboB (
    i { { Timeout A } }
    o { { DialTone A } }
    p { true }
    a { { DialingA=true } }
)

58. HoldCallerAHoldCalledB ConnectedCallerAConnectedCallerB (
    i { { Off-hook B } }
    o { { } }
    p { true }
    a { { } }
)

59. BusyABusyB DialingABusyB (
    i { { Timeout A } }
    o { { DialTone A } }
    p { true }
    a { { DialingA=true } }
)

60. CallingACalledB IdleAIdleB (
    i { { Onhook A } }
    o { { } }
    p { true }
    a { { term=true; IdleA=true; IdleB=true } }
)

61. LimboADialingB DialingADialingB (
    i { { Timeout A } }
    o { { DialTone A } }
    p { true }
    a { { DialingA=true } }
)

```

F16.8


```

62, LimboABusyB DialingABusyB (
    i { {Timeout A} }
    o { {DialTone A} }
    p { true }
    a { DialingA=true }
)

63, IdleABusyB IdleADialingB (
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { DialingB=true }
)

64, DialingABusyB DialingADialingB (
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { DialingB=true }
)

65, DialingALimboB DialingADialingB (
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { DialingB=true }
)

66, BusyABusyB BusyADialingB (
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { DialingB=true }
)

67, BusyALimboB BusyADialingB (
    i { {Timeout B} }
    o { {DialTone B} }
    p { true }
    a { DialingB=true }
)

```

F/6. 9

1 Generate all acyclic paths

1a. Collapse strongly connected components into a single node

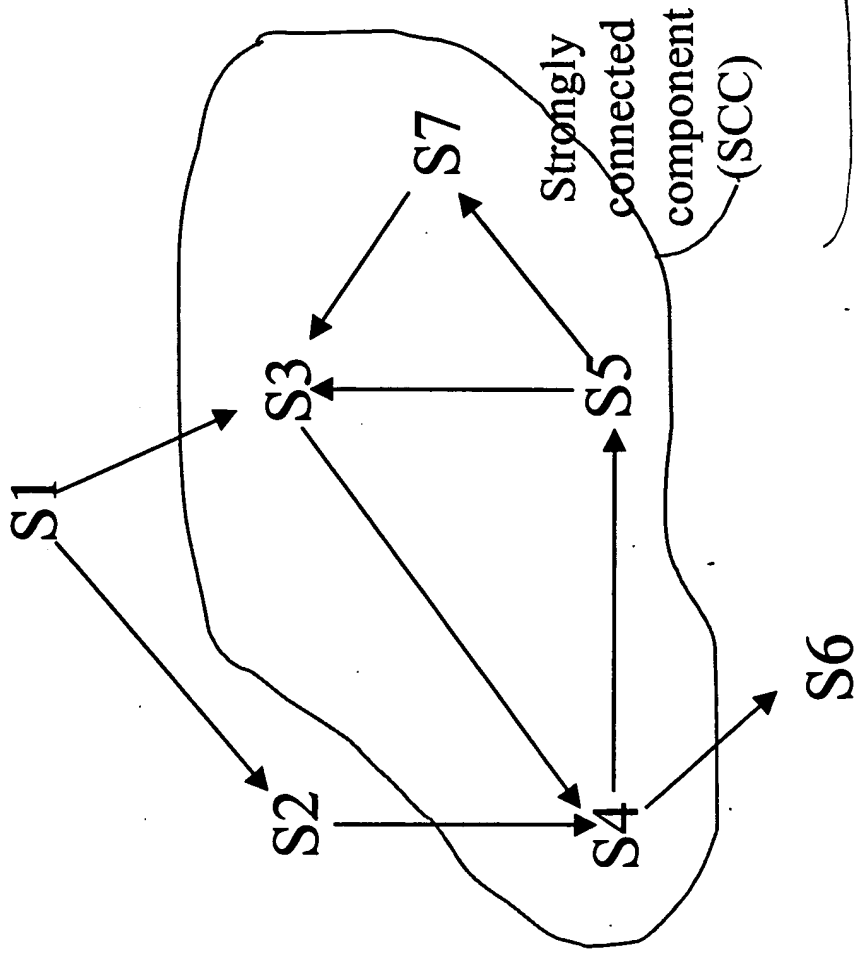
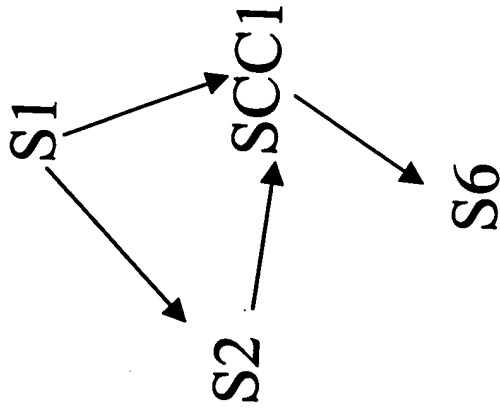


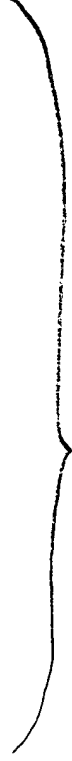
Fig. 10

1 Generate all acyclic paths

1b Determine all paths in DAG:



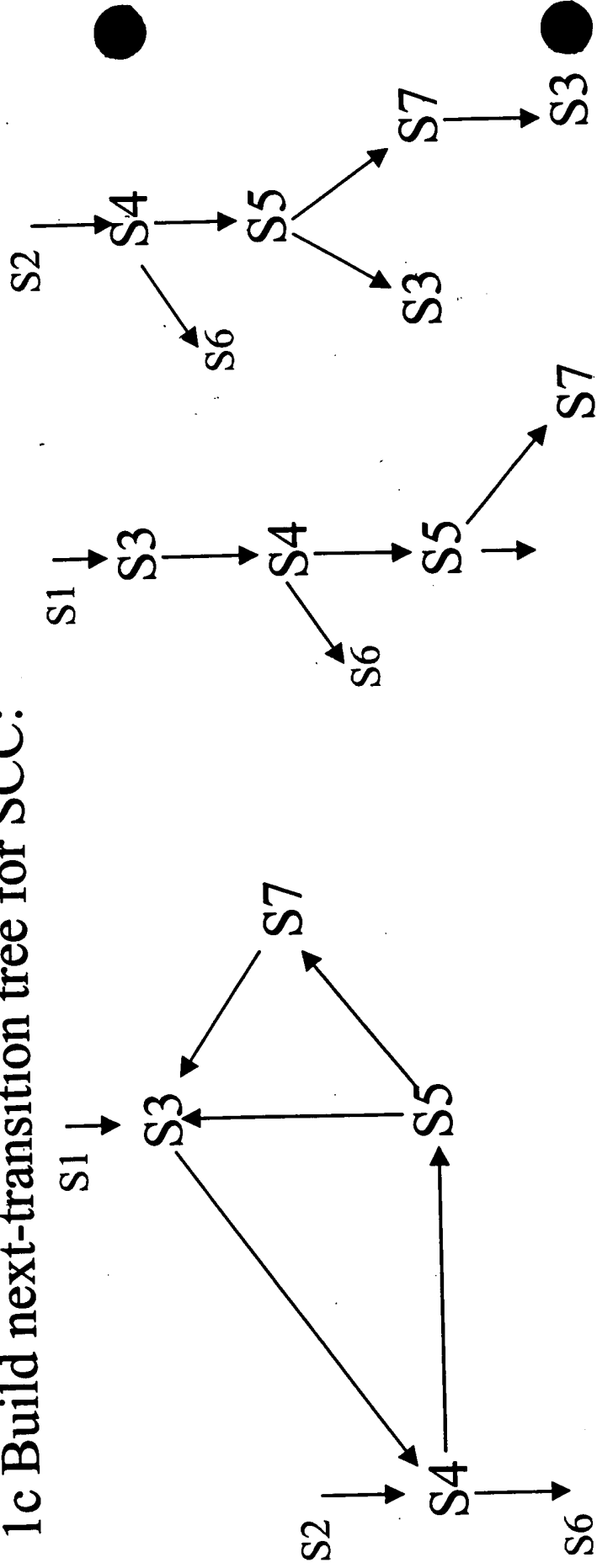
S1, S2, SCC1, S6
S1, SCC1, S6



F/G.11

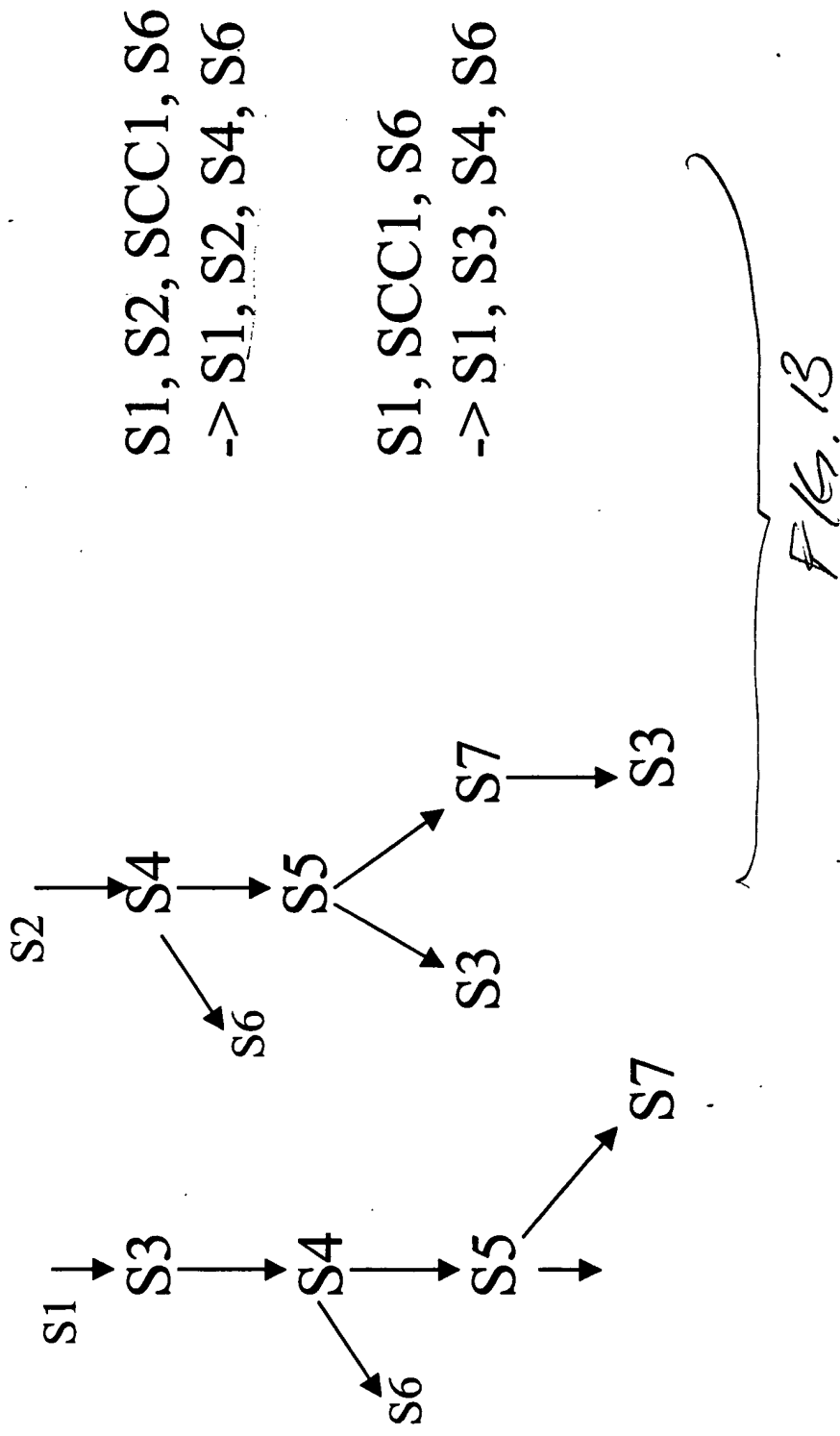
1 Generate all acyclic paths

1c Build next-transition tree for SCC:



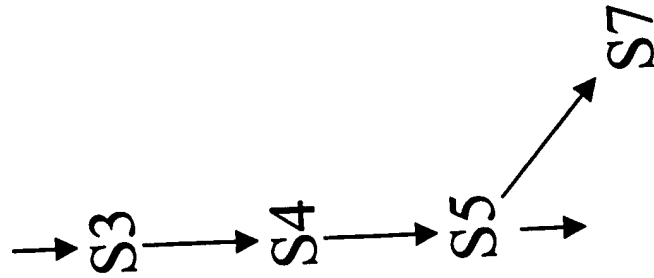
1 Generate all acyclic paths

1d Expand each SCC with all paths through next-transition tree

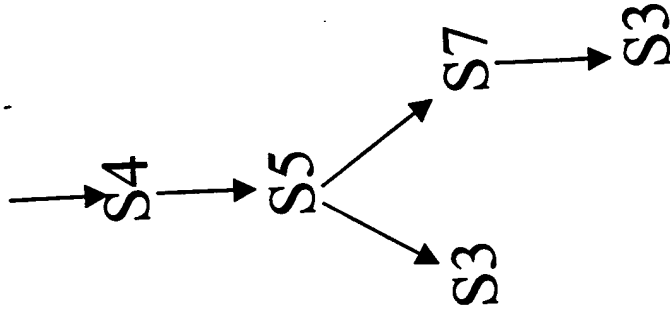


2 Generate all simple cycles

Re-use next-transition tree



S3, S4, S5, S3
S3, S4, S5, S7, S3



S4, S5, S3, S4
S4, S5, S7, S3, S4

FIG. 14

3 Combine the paths and the cycles

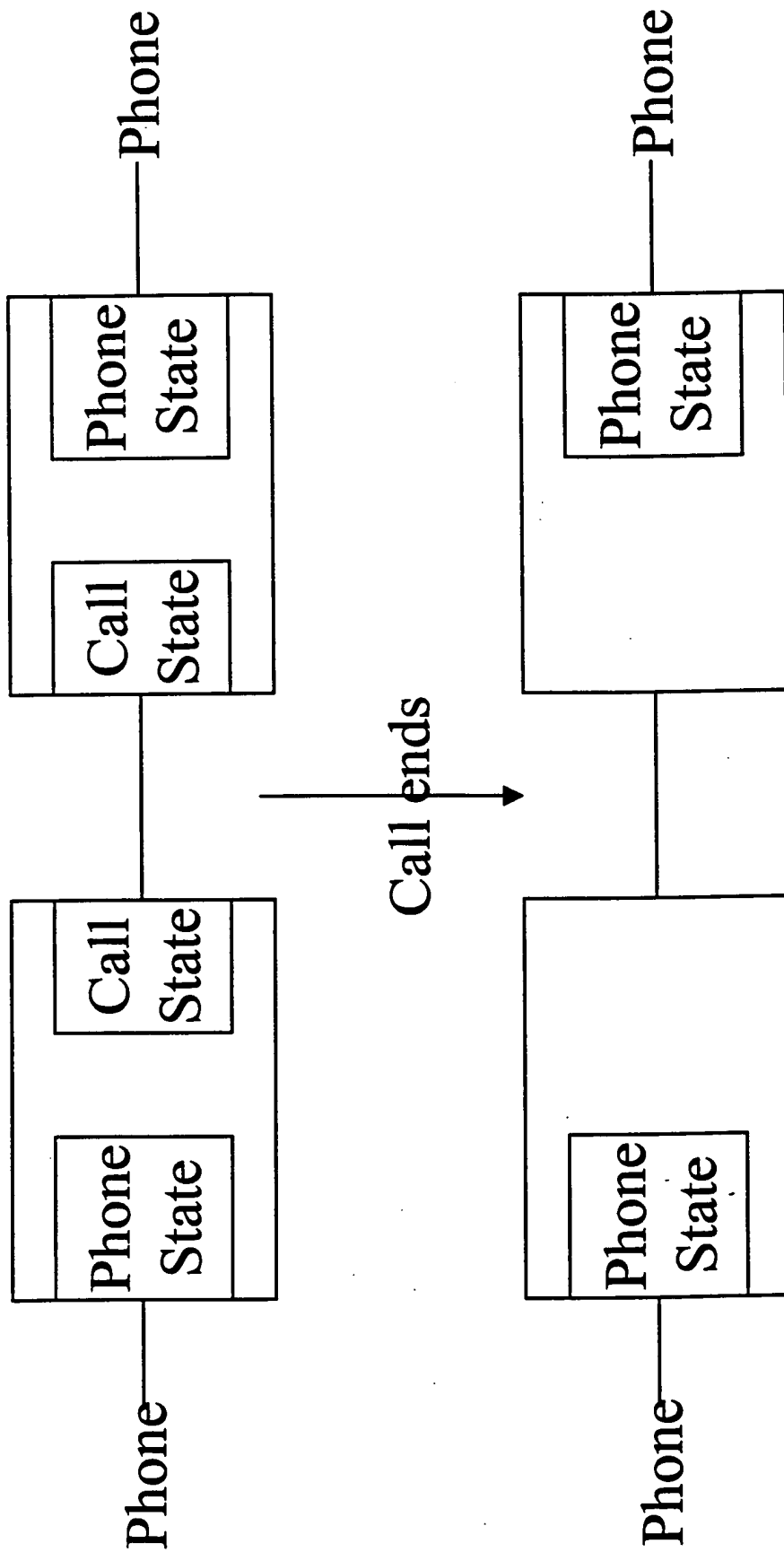
$$\begin{array}{l} S1, S2, S4, S6 \\ S1, S3, S4, S6 \end{array} + \begin{array}{l} S3, S4, S5, S3 \\ S3, S4, S5, S7, S3 \end{array}$$

=

$$\begin{array}{l} S1, S2, S4, S5, S3, S4, S6 \\ S1, S2, S4, S5, S7, S3, S4, S6 \\ S1, S3, S4, S5, S3, S4, S6 \\ S1, S3, S4, S5, S7, S3, S4, S6 \end{array}$$

Fig. 15

Criteria for adequate coverage



FLS 16

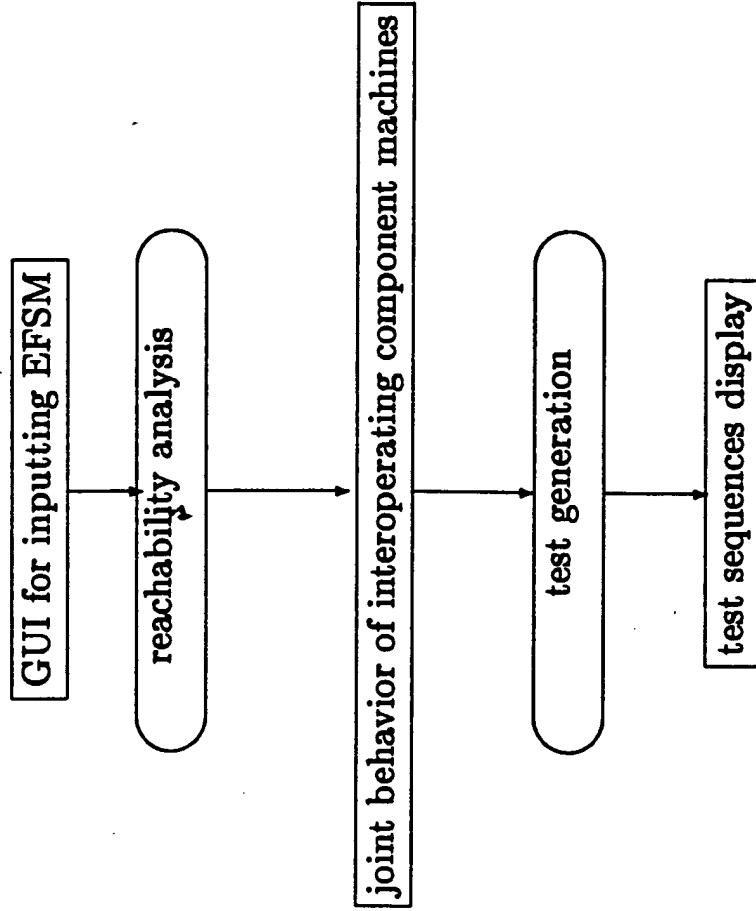


Fig. 17